DOI: 10.58895/ksp/1000162754-11 erschienen in:
Proceedings. 33. Workshop Computational Intelligence - Berlin, 23. – 24. November 2023
DOI: 10.5445/KSP/1000162754 | https://www.ksp.kit.edu/site/books/e/10.5445/KSP/1000162754/

Simulation Model Calibration for Condition Monitoring

Aleksandr Subbotin, Thomas Bartz-Beielstein

Technische Hochschule Köln, Institute for Data Science, Engineering, and Analytics
Steinmüllerallee 1, 51643 Gummersbach, Germany
E-Mail: {aleksandr.subbotin,thomas.bartz-beielstein}@th-koeln.de

Abstract

Condition monitoring is a key component of condition-based and predictive maintenance solutions and has applications in a wide range of industries. However, extracting long-term asset condition information from process data is not a trivial process. The objective of this paper is to present the first steps in developing a condition monitoring solution using a hybrid modeling approach. The paper provides an introduction to condition monitoring and hybrid modeling and focuses on the problem of calibration of first principles based simulation. Several possible approaches to model the calibration coefficients that vary during the process simulation were considered. Our results show that the developed piecewise constant approach, together with the tuned version of the Nelder-Mead optimization algorithm, allows to accelerate the calibration process without sacrificing the simulation error.

1 Introduction

The design life of equipment is often conservative because, in practice, actual operating and environmental conditions may differ significantly from those considered in the design. Therefore, during operation, a remaining life assessment is required to determine the actual remaining life of critical equipment, which may be shorter or longer than the design life [4]. However, extracting long-term asset condition information from process data is not a trivial process. One possible solution could be to use a condition monitoring solution based on a hybrid modelling approach—a combination of a first-principles-based simulation model with machine learning algorithms.

This article presents the results of an ongoing research project with an industry partner, and not all project details can be disclosed. The article is organized as follows. Section 2 provides an introduction to condition monitoring. The hybrid modeling approach and examples of its application to condition monitoring are presented in Section 3. Then, a problem of calibration of the first principles based simulation is introduced in Section 4. Section 5 presents a case study to demonstrate and test the developed model calibration approach. Finally, the conclusions are presented in Section 6.

2 Condition Monitoring

Condition Monitoring (CM) is the process of monitoring the condition of industrial assets (manufacturing equipment, machinery, parts, auxiliary systems and components, etc.) during operation. Condition monitoring is a main part of condition-based and predictive maintenance solutions and has applications in a broad range of industries [1]. In general, the development of a CM solution consists of three main parts: data collection, data exploration and processing, and the development of a CM algorithm [2]. Depending on the industry and field of application, all three parts can vary significantly from solution to solution. The data source for CM can be either specially designed and installed sensors [3] or the existing infrastructure used for process monitoring [**?**]. The basic idea behind data-driven condition monitoring is that it is possible to extract some patterns and trends—condition indicators—from a large amount of collected data and infer the deterioration status of equipment for which there are not available or do not exist condition monitoring sensors. Data-driven condition monitoring relies on various data sources and types of measurements acquired during equipment operation and uses various data mining techniques and algorithms [2].

3 Hybrid Modeling

Hybrid modelling is a combination of two paradigms: first principles-based and data driven models into a single architecture (Fig. 1). First-principles (physics-based) models are based on formalized expert knowledge of a problem, including design data, material properties, etc. In contrast, data-driven methods rely only on collected data. Hybrid modelling already has a portfolio

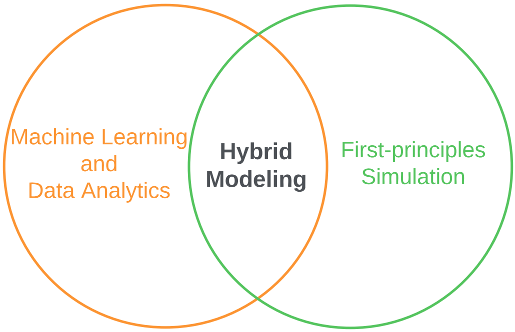


Figure 1: Hybrid modeling is the fusion of two worlds: machine learning and first-principles based simulation.

of applications in the process industry [6] and several examples of applications to condition monitoring. Leturiondo, et al. [8] use hybrid modeling to monitor the condition of rolling bearings. Gálvez, et al. [**?**] use hybrid modeling for condition monitoring of the heating, ventilation, and air conditioning (HVAC) system in passenger trains. In both works, the authors proceed from the hypothesis that, due to scheduled maintenance and service of equipment, the real measurements, collected by sensors located in the real system, contain very limited information about the degradation of elements, especially in the late stages of degradation. Therefore, in both studies, physics-based models are used to generate synthetic data for operation with known degradation levels and equipment failures. In our study, we would like to propose and test a different way of condition monitoring by using a hybrid modelling approach. We assume that in real measurements it is possible to track the degradation process of the equipment by using detailed and calibrated physics-based simulation of the process. The calibration coefficients of the physics-based model can possibly be used as condition indicators. For this purpose the process of physics-based model calibration should be relatively fast because we plan to trace the changes of physics-based model calibration coefficients during the whole lifetime of the equipment. If our hypothesis is successful, the next step would be to create a data-driven model that could extract condition indicators from process data without the use of simulation and optimization.

4 Simulation Calibration

One of the important steps in the development of a hybrid model is the calibration of the physics-based model (simulation) to better match the industrial data (real measurements). Model calibration is the manual or automated process of estimating and adjusting model parameters (calibration coefficients) by fitting the model output to real data.

The simulation calibration process consists of the following steps (Fig. 2). In the first step, the equipment design data and actual process data (measurements) are prepared and fed into the simulation. The second step is the execution of the process simulation. In the third step, the parameters calculated in the simulation are compared with the real measurements. The simulation error is calculated and passed to the optimization algorithm. The goal of the optimization algorithm in the fourth step is to minimize the simulation error by finding optimal calibration coefficients. The process is repeated until the desired accuracy is achieved or the number of simulations is exhausted. The choice of an effective optimization algorithm for model calibration is not obvious and requires considerable experience and some experimentation.

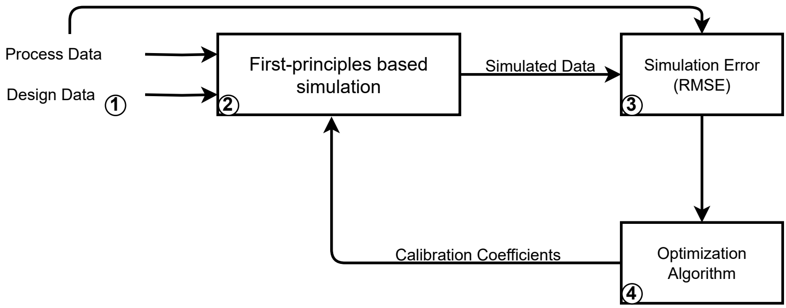


Figure 2: Flowchart of the Simulation Calibration Process

4.1 Calibration Coefficients

The process simulation has a set of calibration coefficients. From the available coefficients, we have selected two that could potentially be used as condition indicators. Since they are just coefficients without units, we can just label them as Calibration Coefficient 1 (CC1) and Calibration Coefficient 2 (CC2). The Calibration Coefficient 1 does not change during the one process simulation. The Calibration Coefficient 2, on the other hand, decreases during the process simulation (Fig. 3), but can be modeled as a constant for simplification. Modeling CC2 as a constant throughout the process simulation leads to oversimplification and is suitable for approximate estimation, but doesn't fit the purpose of what we are working on.

Due to the complex behavior of the CC2, two other calibration approaches were considered.

Approach 1 (exponentially decaying curve): In theory, the behavior of CC2 can be modeled with an exponentially decaying curve (Fig. 3) described by the equation $CC2 = A \cdot \exp\left(-\frac{t}{T}\right) + B$, where t is the simulation time; A, B and T are unknown parameters. Unfortunately, attempts to determine optimal values for the parameters A, B and T have not been successful because this approach requires too many time-consuming simulations.

Approach 2 (piecewise constant): Because of the difficulties we encountered in determining the coefficients of the exponential curve, we developed a different

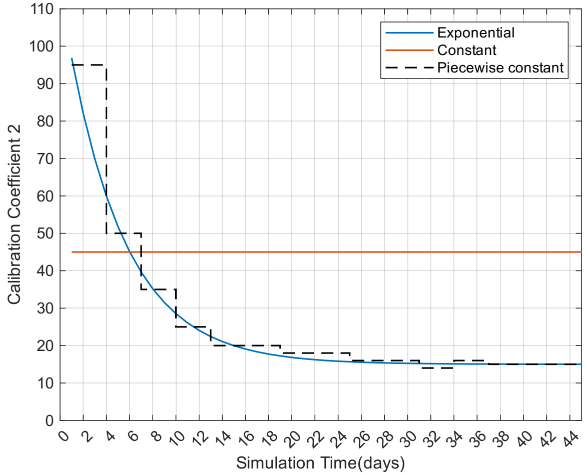


Figure 3: Three ways to model CC2: 1–Constant throughout the simulation; 2–Exponentially decaying curve; 3–Piecewise constant

approach. The simulation is divided into many small steps (overlapping windows) in which the CC2 is modeled as a constant. Then, using an optimization algorithm, the optimal value of the CC2 is calculated sequentially for each step. In this approach, the CC1 is optimized only on the first step.

4.2 Optimization Algorithm

The process of calibration of the simulation model can be considered as simulation-based optimization problem or Derivative-Free Optimization (DFO) problem. Simulation optimization is a very broad topic that involves the use of algorithms that come from many different fields, have connections to many different disciplines, and have been used in many practical applications [7]. DFO can be considered as a sub-field of simulation optimization. Most algorithms in DFO are specifically designed to consider that function evaluations or simulations are expensive.

The optimization objective is to minimize the difference between the real data and the simulated data. In our case, the Root Mean Square Error (RMSE) between the simulated parameter and the actual (observed) measurement used as a cost function. Since every objective function evaluation requires a time-consuming process simulation and gradient information is not available, DFO algorithms are best suited for our task. The list of DFO algorithms built into the MATLAB environment we use, including the Statistics and Machine Learning Toolbox, the Optimization Toolbox, and the Global Optimization Toolbox, is given in Table 1.

Table 1: List of DFO methods built into MATLAB and available in MATLAB Toolboxes

Algorithm	Matlab Function
Nelder-Mead Simplex Method	fminsearch
Golden Section Search	fminbnd
Pattern Search	patternsearch
Surrogate Optimization	surrogateopt
Genetic Algorithm	ga
Particle Swarm Solver	particleswarm
Simulated Annealing	simulannealbnd
Bayesian Optimization	bayesopt

5 Experiment

The goal of our work is not a detailed benchmarking of algorithms, but the selection of the most efficient optimization algorithm for our concrete problem. We formulated several requirements for the optimization algorithm:

- The calibration coefficients have physical bounds, e.g., from 0% to 100%, and the simulation cannot take values outside these limits. For this reason, the optimization algorithm must be constrained.
- Detailed simulation takes time, so the algorithm must use a limited number of function evaluations (as few as possible).
- Since we have several coefficients to optimize, the optimization algorithm should support multi-variable optimization.

The MATLAB software package contains several algorithms suitable for our problem. We compare three of them: a constrained version of the Nelder-Med simplex method, Pattern search and Bayesian optimization. Nelder-Mead and Pattern Search are widely used direct search methods [10]. In the MATLAB implementation, the Nelder-Mead algorithm does not support constraints. However, there is a popular community version of the algorithm with constraints. Bayesian optimization is a global optimization algorithm recently become extremely popular for tuning hyperparameters in machine learning models [11].

5.1 Experiment Setup

For the experiment, we used the industrial process simulation with a total duration of 85 days. According to the developed piecewise constant approach, the simulation is divided into 28 steps of 5 days each (window size is 5 days with an overlap of 2 days). All optimization algorithms are set to the same maximum number of iterations, 30 for the first step (window) and 15 for all subsequent steps. For the Nelder–Mead and Pattern Search algorithms, the starting point is the optimal solution from the previous step. The optimization constraints are the same for all algorithms.

For the Nelder-Mead algorithm, we carried out two experiments, one with the default settings and the other with a modified tolerance for both the objective function value and the variable.The tolerance settings of the Pattern Search algorithm are equivalent to the tuned version of the Nelder-Mead algorithm, and for Bayesian Optimization, there are no tolerance settings. The settings of the experiments are summarized in the Table 2.

5.2 Experiment Results

All three optimization algorithms solved the optimization problem, but with a significantly different number of objective function evaluations and a slightly different RMSE values after optimization. Since we have 28 steps in our

Table 2: Experiment Setup

Short Name	Algorithm	Tolerance
NM_1	Nelder-Mead	Default: 1e-4
NM_2	Nelder-Mead	TolFun: 1e-03 TolX: 0.1
BO	Bayesian Optimization	Does not support tolerance settings
PS	Pattern Search	TolFun: 1e-03 StepTolerance: 0.1 MeshTolerance: 0.1

proposed piecewise constant approach, the median RMSE value is used to compare the optimization algorithms. Figure 4 shows the number of evaluations of the objective function (simulations) that are used by each of the algorithms at each of the 28 steps. The Nelder-Mead algorithm with the default setting uses slightly more function evaluations than it was limited to use. However, by tuning the optimization tolerance, we were able to significantly reduce the number of function evaluations while keeping the median RMSE at the same level. This is illustrated in Figure 5, which shows the total number of function evaluations and the median RMSE value over 28 steps. The Pattern Search algorithm, with the same tolerance settings as the tuned Nelder-Mead algorithm, required more objective function evaluations and has a slightly worse RMSE values. As expected, Bayesian optimization without tolerance settings uses the entire limit of objective function evaluations, but does not show better RMSE values. The results of the experiment are summarized in Table 3. It should be noted that we also found that the constrained version of the Nelder-Mead algorithm can have convergence problems when the solution is very close to the boundary, which is not a problem for Bayesian optimization and Pattern search. The resulting values of CC2 after the optimization process are shown in Figure 6.

Table 3: Experiment Result

Algorithm Short Name	Total Number of Function Evaluations	Optimal Value CC1	Median Simulation Error
NM_1	462	1.087	0.0025
NM_2	**218**	1.087	**0.0025**
BO	435	1.094	0.0028
PS	291	1.100	0.0031

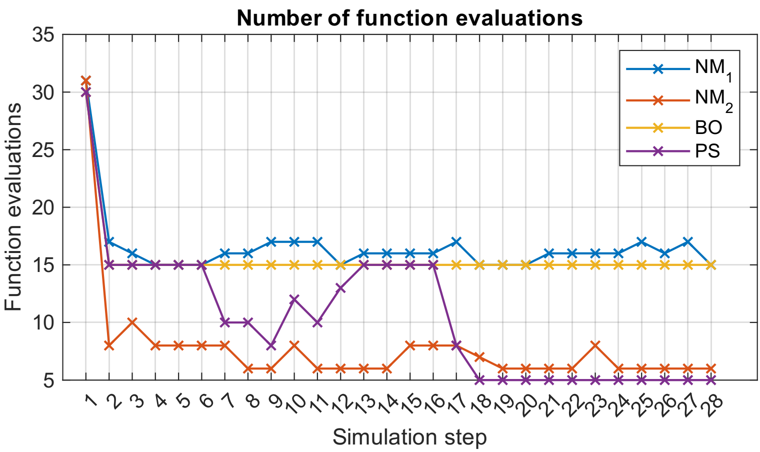
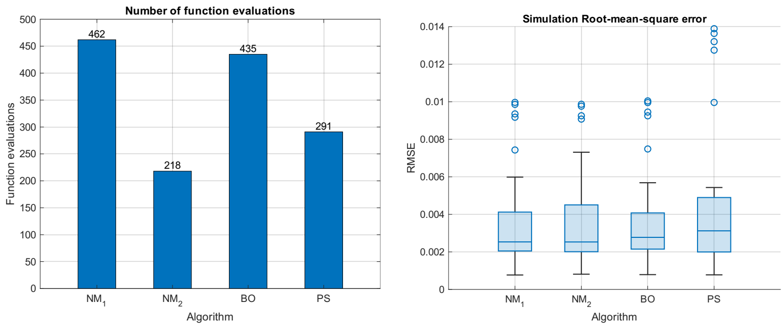


Figure 4: Number of function evaluations for each calibration step. Tuned version of the Nelder-Mead algorithm (NM_2) uses less objective function evaluations, especially in the first 17 steps, than other optimization methods.

6 Conclusion

This paper presents the first steps in the development of a condition monitoring solution using hybrid modeling. In this phase, we considered several possible ways to model calibration coefficients that vary during the process simulation. To address this issue, the piecewise constant approach was developed and then tested with three different optimization algorithms and different optimization tolerance settings. The experimental results show that the simulation calibration process can be significantly accelerated by tuning the tolerance parameter



(a) Number of Function Evaluations

(b) Simulation RMSE after optimization

Figure 5: Experiment Results. a) Total number of function evaluations performed by each algorithm; b) Simulation RMSE statistics over 28 calibrations steps for each algorithm

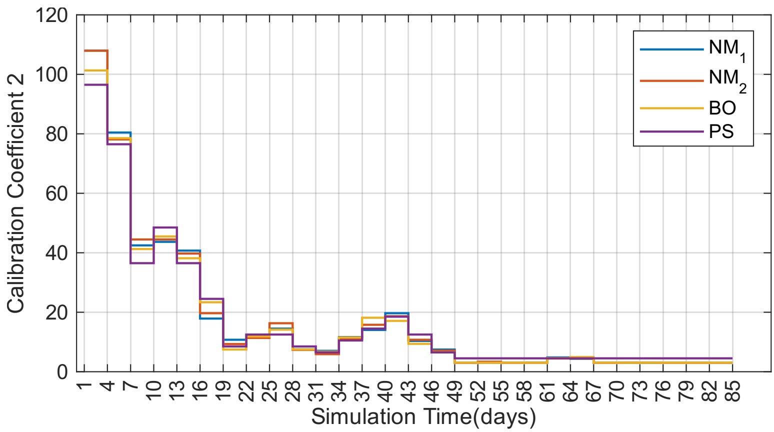


Figure 6: Resulting CC2 values after calibration procedure. The values are only slightly different from each other, but the number of function evaluations that are used is completely different, see Fig. 5

of the optimization algorithm without sacrificing the simulation error. The best results were obtained using the tuned version of the Nelder-Mead algorithm, but the optimal balance between optimization speed and simulation error needs to be further investigated. The next step is to use a developed simulation calibration approach to determine the potential of using the calibration coefficients as condition indicators in a condition monitoring solution.

References

[1] J. Naik, A. Acharya, J. Thaker “Revolutionizing condition monitoring techniques with integration of artificial intelligence and machine learning” In: *Materials Today: Proceedings* In Press https://doi.org/10.1016/j.matpr.2023.08.262 2023.

[2] R. Qi, J. Zhang, K. Spencer “A Review on Data-Driven Condition Monitoring of Industrial Equipment” In: *Algorithms* 16 Pages 1-9 https://doi.org/10.3390/a16010009 2023.

[3] P. Strauß, M. Schmitz, R. Wöstmann, J. Deuse “Enabling of Predictive Maintenance in the Brownfield through Low-Cost Sensors, an IIoT-Architecture and Machine Learning” In: *IEEE International Conference on Big Data (Big Data)* Pages 1474-1483 https://doi.org/10.1109/BigData.2018.8622076 2018.

[4] I. Animah, M. Shafiee “Condition assessment, remaining useful life prediction and life extension decision making for offshore oil and gas assets” In: *Journal of Loss Prevention in the Process Industries* 53 Pages 17-28 https://doi.org/10.1016/j.jlp.2017.04.030 2018.

[5] P. Zürcher, S. Badr, S. Knüppel, H.Sugiyama, “Data-driven equipment condition monitoring and reliability assessment for sterile drug product manufacturing: Method and application for an operating facility” In: *Chemical Engineering Research and Design* 188, Pages 301-314 https://doi.org/10.1016/j.cherd.2022.09.005 2022.

[6] J. Glassey, M. von Stosch “Hybrid modeling in process industries” CRC Press https://doi.org/10.1201/9781351184373 2018.

[7] A. Satyajith, S. Nikolaos V., S. Bikram, B. Scott J. “Simulation optimization: a review of algorithms and applications” In: *Annals of Operations Research* Pages 351–380 2015.

[8] U. Leturiondo, M. Mishra, D. Galar, O. Salgado “Synthetic data generation in hybrid modelling of rolling element bearings”. In: *Insight*

- *Non-Destructive Testing and Condition Monitoring* Pages 395-400 2015.

[9] A. Gálvez, J. Rubio, D. Seneviratne, A. Gonzalez, A. Jimenez, U. Martinez-de-Estarrona, D. Galar, E Juuso “Hybrid Models and Digital Twins for Condition Monitoring: HVAC System for Railway”. In: *SNE Simulation Notes Europe* Pages 121–126 2021.

[10] J. Larson, M. Menickelly, S. M. Wild “Derivative-free optimization methods”. In: *Acta Numerica* 28 Pages 287-404 https://doi.org/10.1017/s0962492919000060 2019.

[11] P. I. Frazier “A Tutorial on Bayesian Optimization”. In: *arXiv 1807.02811* https://arxiv.org/pdf/1807.02811.pdf 2018.